1	What is claimed is:
1	1. A processor comprising:
2	a cache;
3	an execution unit to execute an instruction having an operand indicating a monitor
4	address;
5	a bus controller to assert a preventative signal in response to receiving a memory
6	access attempting to gain sufficient ownership of a cache line associated with
7	said monitor address to allow a modification of said cache line without
8	generation of another transaction indicative of the modification.
1	
1	2. The processor of claim 1 wherein said cache is an L1 cache and wherein said
2	processor further comprises an L2 cache.
1	
1	3. The processor of claim 2 wherein the cache line associated with said monitor address
2	is flushed from the L1 cache and the L2 cache in response to said instruction.
1	
1	4. The processor of claim 1 wherein said bus controller is to generate a bus cycle in
2	response to the instruction, the bus cycle to eliminate any ownership of said cache
3	line by another processor that would allow modification of said cache line without
4	generation of another transaction indicative of modification of the cache line.
1	
1	5. The processor of claim 4 wherein said bus cycle is a read and/or invalidating bus
2	cycle.
1	

1	6. The processor of claim 5 further comprising a monitor coupled to said bus controller
2	to monitor bus transactions for a transaction indicative of a write to the monitor
3	address, and to signal a monitor event in response to the transaction indicative of the
4	write to the monitor address.
1	
1	7. The processor of claim 2 further comprising:
2	a plurality of write combining buffers between said L1 cache and said L2 cache;
3	a snoop port for said plurality of write combining buffers;
4	a monitor coupled to said L1 cache and coupled to the snoop port to monitor
5	memory access cycles from the execution unit and from the snoop port.
1	
1	8. An apparatus comprising:
2	a bus controller having a plurality of bus cycle information lines;
3	programmable memory access detection logic coupled to a bus, said
4	programmable memory access detection logic comprising a storage location to
5	store a monitor address specified by an instruction and comparison logic
6	having inputs coupled to said storage location and said plurality of bus cycle
7	information lines and a comparison logic output;
8	coherence logic coupled to receive said monitor address, that in response to the
9	instruction is to generate a read and/or invalidate transaction for a cache line
10	associated with said monitor address.
1	
1	9. The apparatus of claim 8 wherein said programmable memory access detection logic

42390.P13635 -30-

2	comprises write detection logic.
1	
1	10. The apparatus of claim 9 further comprising:
2	hit generation logic coupled to said storage location and said bus controller,
3	wherein said hit generation logic has an output hit signal externally available
4	to couple to a system bus.
1	
1	11. The apparatus of claim 8 wherein said bus controller is to generate, in response to the
2	instruction, a bus cycle chosen from a set consisting of:
3	a bus read line invalidate of the cache line associated with said monitor address;
4	a bus write line invalidate of the cache line associated with said monitor address.
1	
1	12. The apparatus of claim 8 wherein said read and/or invalidate transaction is to ensure
2	that no other processor caches include said cache line associated with said monitor
3	address in a modified or exclusive state.
1	
1	13. The apparatus of claim 8 wherein said instruction is a part of a first thread, further
2	comprising:
3	suspend logic responsive to a second instruction in said first thread to suspend
4	said first thread;
5	resume logic responsive to said comparison logic output to resume said first
6	thread.

1

1	14. The apparatus of claim 13 further comprising:
2	partition/anneal logic to anneal and partition resources responsive to respectively
3	suspension and resumption of said first thread.
1	
1	15. A method comprising:
2	performing a first bus transaction to eliminate ownership by other agents of a
3	cache line associated with a monitor address specified by an instruction;
4	asserting a preventative signal in response to a second bus transaction attempting
5	to gain ownership of said cache line associated with the monitor address.
1	
1	16. The method of claim 15 wherein performing the first bus transaction comprises:
2	preventing any system processor cache from storing said cache line associated
3	with said monitor address in a modified or exclusive state.
1	
1	17. The method of claim 15 wherein performing the first bus transaction comprises
2	performing an invalidating transaction.
1	
1	18. The method of claim 15 wherein performing the first bus transaction comprises
2	performing a read transaction.
1	
1	19. The method of claim 15 wherein asserting the preventative signal comprises asserting
2	a hit signal in response to a transaction which could result in a bus agent gaining
3	ownership of the cache line associated with said monitor address.
1	

1	20. The method of claim 15 wherein said monitor address is an operand of said
2	instruction.
1	
1	21. The method of claim 15 further comprising flushing said cache line from a plurality
2	of processor caches in a processor that executes said instruction.
1	
1	22. The method of claim 15 further comprising:
2	suspending execution of a first thread of which the instruction is a part, in
3	response to a second instruction;
4	resuming execution of said first thread in response to detection of a memory
5	access to the monitor address.
1	
1	23. The method of claim 22 wherein suspending execution of the first thread further
2	comprises:
3	relinquishing a plurality of thread partitionable resources associated with said firs
4	thread.
1	
1	24. A system comprising:
2	a bus;
3	a first processor having a first cache;
4	a second processor having a second cache, said second processor comprising:
5	a monitor to monitor transactions from the first processor on the bus to
6	detect a memory access to a monitor address specified by an

42390.P13635 -33-

7	instruction executed by said second processor;
8	coherence logic to generate a bus transaction to prevent said first cache
9	from owning a cache line associated with said monitor address.
1	
1	25. The system of claim 24 wherein said second processor further comprises:
2	hit generation logic to generate a hit signal in response to a read
3	transaction to said cache line associated with said monitor address.
1	
1	26. The system of claim 24 wherein said second processor is to flush said cache line
2	associated with said monitor address from the second cache in response to said
3	instruction.
1	
1	27. The system of claim 24 wherein said bus transaction to prevent said first cache from
2	owning said cache line associated with said monitor address is a read transaction.
1	
1	28. The system of claim 24 wherein said bus transaction prevents the first cache from
2	holding said cache line associated with said monitor address in a modified or
3	exclusive state.
1	
1	29. The system of claim 24 wherein said second processor further comprises thread
2	suspension logic to suspend a first thread of which said instruction is a part until an
3	access to said cache line associated with said monitor address occurs.
1	

5

- 30. The system of claim 29 wherein said second processor further comprises partitioning
- and annealing logic to relinquish resources associated with said first thread when said
- 3 first thread is suspended and to re-partition resources to accommodate said first thread
- 4 when said first thread is resumed.